White Paper

IMPLEMENTING FAST VP AND STORAGE TIERING FOR ORACLE DATABASE 11*g* AND EMC SYMMETRIX VMAX

Abstract

As the need for more information continues to explode, businesses are forced to deal with an ever-increasing demand for storage and at the same time, a requirement to keep cost to a minimum. One way to satisfy both needs is to effectively use low-cost SATA drives, high-performance Fibre Channel drives, and Enterprise Flash Drives to provide high performance for mission-critical applications and cost-effective storage for non-critical applications. Matching business needs with various drive types is known as “tiering.” This white paper describes how to implement storage tiering using EMC® Symmetrix VMAX™ Enhanced Virtual LUN Technology and Fully Automated Storage Tiering for Virtual Pools, or FAST VP.

April 2011

Part Number h8131.2

Table of Contents

Executive summary

The EMC® Symmetrix VMAX™ series with Enginuity is the newest addition to the Symmetrix® product family. Built on the strategy of simple, intelligent, modular storage, it incorporates a new scalable Virtual Matrix™ interconnect that connects all shared resources across all VMAX Engines, allowing the storage array to grow seamlessly and cost-effectively from an entry-level configuration into the world's largest storage system. The Symmetrix VMAX provides improved performance and scalability for demanding enterprise storage environments while maintaining support for EMC's broad portfolio of platform software offerings.

EMC Symmetrix VMAX delivers enhanced capability and flexibility for deploying Oracle databases throughout the entire range of business applications, from mission-critical applications to test and development. In order to support this wide range of performance and reliability at minimum cost, Symmetrix VMAX arrays support multiple drive technologies that include Enterprise Flash Drives (EFDs), Fibre Channel (FC) drives, both 10k rpm and 15k rpm, and 7,200 rpm SATA drives. In addition, various RAID protection mechanisms are allowed that affect the performance, availability, and economic impact of a given Oracle system deployed on a Symmetrix VMAX array.

As companies increase deployment of multiple drive and protection types in their high-end storage arrays, storage and database administrators are challenged to select the correct storage configuration for each application. Often, a single storage tier is selected for all data in a given database, effectively placing both active and idle data portions on fast FC drives. This approach is expensive and inefficient, because infrequently accessed data will reside unnecessarily on high-performance drives.

Alternatively, making use of high-density low-cost SATA drives for the less active data, FC drives for the medium active data, and EFDs for the very active data enables efficient use of storage resources, and reduces overall cost and the number of drives necessary. This, in turn, also helps to reduce energy requirements and floor space, allowing the business to grow more rapidly.

Database systems, due to the nature of the applications that they service, tend to direct the most significant workloads to a relatively small subset of the data stored within the database and the rest of the database is less frequently accessed. The imbalance of I/O load across the database causes much higher utilization of the LUNs, holding the active objects in a phenomenon known as LUN access "skewing." However, in most cases LUNs have some unallocated and therefore idle spaces, or a combination of hot and cold data due to a mix of different database objects. Such differences in the relative utilization of the space inside each LUN are referred to as sub-LUN "skewing."

While the use of multiple storage tiers can be managed manually by DBAs placing the appropriate database objects in their right tier, this can become cumbersome given the growing complexity of applications and the fluctuations of access frequency to data over time.

Enginuity 5874 introduced Fully Automated Storage Tiering (FAST) as a method to address changes in LUN access skewing. FAST operates on standard (non-VP) Symmetrix addressable devices. It automatically and seamlessly moves the storage behind the controlled LUNs to the appropriate storage tier, based on user policy and LUN activity. Enginuity 5875 introduced FAST for Virtual Pools (FAST VP) as a method to address changes in sub-LUN access skewing. FAST VP is based on Virtual Provisioning™ and operates on thin Symmetrix devices. It automatically and seamlessly moves portions of the LUN to the appropriate storage tiers, based on user policy and the sub-LUN activity. Due to its finer granularity, FAST VP is more efficient in utilizing the capacity of the different storage tiers, and more responsive to changes in workload patterns than even the most diligent DBA. FAST VP also adapts readily to configurations in which, due to host striping, the workload is evenly distributed across many LUNs (like Oracle Automatic Storage Management, or ASM). Rather than having to move all the LUNs as a group between storage tiers, FAST VP operates appropriately on small portions in each LUN, moving them to the storage tier that best matches their workload needs.

FAST VP preserves Symmetrix device IDs, which means there is no need to change filesystem mount points, volume manager settings, database file locations, or scripts. It also maintains any TimeFinder® or SRDF® business continuity operations even as the data migration takes place.

By optimizing data placement of active LUNs and sub-LUNs to the storage tier that best answers their needs, FAST VP helps maximize utilization of Flash drives, increase performance, reduce the overall number of drives, and improve the total cost of ownership (TCO) and ROI. FAST VP enables users to achieve these objectives while simplifying storage management.

Audience

This white paper is intended for Oracle database administrators, storage administrators and architects, customers, and EMC field personnel who want to understand the implementation of storage tiering in a Symmetrix VMAX environment.

Introduction

Evolution of storage tiering

Storage tiering has evolved over the past several years from a completely manual process to the automatic process it is today.

Manual storage tiering

Manual storage tiering is the process of collecting performance information on a set of drives and then manually placing data on different drive types based on the performance requirement for that data. This process is typically very labor-intensive and does not dynamically adjust as the load on the application increases or decreases over time.

Fully Automated Storage Tiering (FAST)

FAST was introduced in 2009 and is based on virtual LUN (VLUN) migration for standard devices. FAST allows administrators to define policies and priorities that govern what data resides in each storage tier and can automatically make data placement decisions without human intervention. FAST is a major step forward in data management automation, but it is limited to moving entire LUNs from one tier to another. Even if only a small amount of the data on the LUN is active then inactive data is also migrated, consuming valuable space in the higher-performance tier.

Fully Automated Storage Tiering for Virtual Pools (FAST VP)

FAST VP monitors the performance of a LUN at fine granularity and moves only a small number of Symmetrix tracks between storage tiers. FAST VP automates the identification of sub-LUN data for the purposes of relocating it across different performance/capacity tiers within an array.

Figure 1 shows an example of storage tiering evolution from a single tier to sub-LUN tiering. Although the image shows FAST VP operating on two tiers alone, in most cases tiering strategy is still best optimized for cost/performance using a three-tier approach.

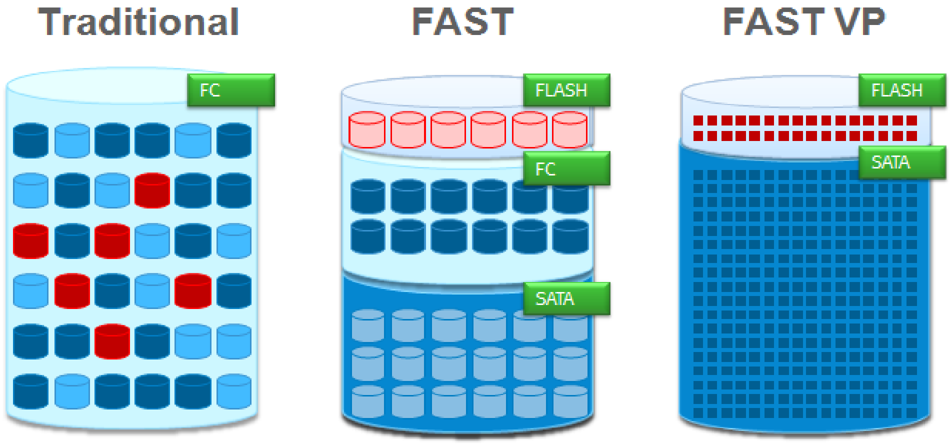


Figure 1. Evolution of storage tiering

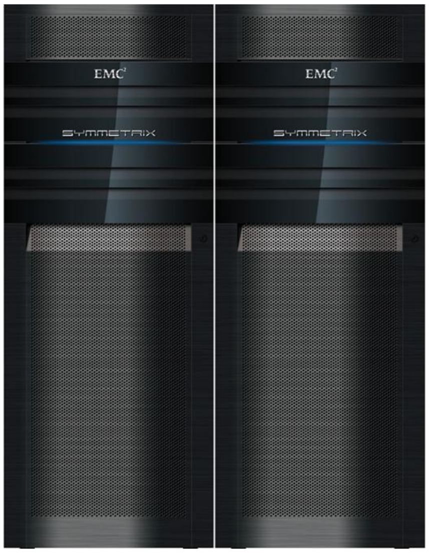
Virtual LUN VP Mobility and FAST VP

This white paper demonstrates best practices for using Symmetrix VMAX Virtual LUN VP Mobility (VLUN VP) for thin device, Virtual Provisioning, and FAST VP technologies with Oracle databases. VLUN VP and FAST VP are complementary technologies. With FAST VP, data movement at the sub-LUN level is automatically and seamlessly executed by the FAST VP controller in the storage array. It evaluates and moves allocated thin device extents across multiple storage tiers as needed, in compliance with a FAST VP policy and based on changes in the workload. VLUN VP Mobility is initiated by the user and then executed seamlessly in the storage array, effectively performing a one-time movement of all allocated thin device extents to a single target storage tier within a given FAST policy. This manual “override” option helps FAST users respond rapidly to changing performance requirements or unexpected events. Virtual Provisioning is the base storage layout on which FAST VP operates.

Products and features overview

Symmetrix VMAX series with Enginuity

Symmetrix VMAX, the newest member of the Symmetrix family, is a revolutionary storage system purpose-built to meet all data center requirements as seen in Figure 2. Based on the Virtual Matrix Architecture™ and new Enginuity capabilities, Symmetrix VMAX scales performance and capacity to unprecedented levels, delivers continuous operations, and greatly simplifies and automates the management and protection of information.



- 1 – 8 redundant VMAX Engines
- Up to 2.1 PB usable capacity
- Up to 128 FC FE ports
- Up to 64 FICON FE ports
- Up to 64 Gig-E / iSCSI FE ports
- Up to 1 TB global memory (512 GB usable)
- 48 – 2,400 drives
- Enterprise Flash Drives 200/400 GB
- FC drives 146/300/450 GB 15k rpm
- FC drives 300/450/600 GB 10k rpm
- SATA drives 2 TB 7.2k rpm

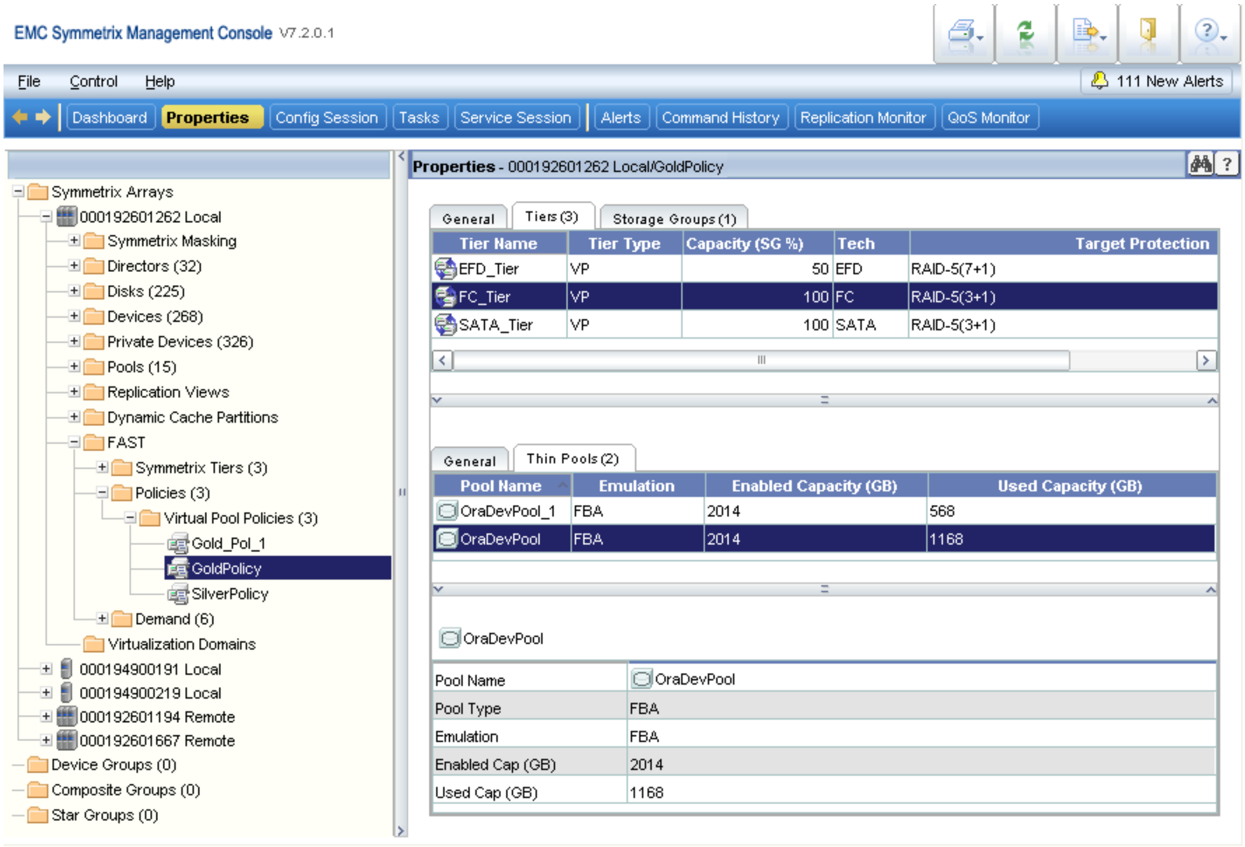
Figure 2. The Symmetrix VMAX platform

The Symmetrix VMAX design is based on individual engines with redundant CPU, memory, and connectivity on two directors for fault tolerance. VMAX Engines connect to and scale out through the Virtual Matrix Architecture, which allows resources to be shared within and across VMAX Engines. To meet growth requirements, additional VMAX Engines can be added nondisruptively for efficient and dynamic scaling of capacity and performance that is available to any application on demand.

Symmetrix Management Console (SMC)

Many large enterprise data centers have stringent change control processes that ensure reliable execution of any modification to their IT infrastructure. Often changes are implemented using scripts that are fully documented, have been thoroughly reviewed, and can be consistently executed by all storage administrators. An alternative to scripts is Symmetrix Management Console (SMC).

SMC, as shown in Figure 3, is a GUI that allows storage administrators to easily manage Symmetrix arrays. SMC can be run on an open systems host connected directly or remotely to a Symmetrix VMAX that requires management or monitoring.



Properties - 000192601262 Local/GoldPolicy

Tier Name	Tier Type	Capacity (SG %)	Tech	Target Protection
EFD_Tier	VP	50	EFD	RAID-5(7+1)
FC_Tier	VP	100	FC	RAID-5(3+1)
SATA_Tier	VP	100	SATA	RAID-5(3+1)

Pool Name	Emulation	Enabled Capacity (GB)	Used Capacity (GB)
OraDevPool_1	FBA	2014	568
OraDevPool	FBA	2014	1168

OraDevPool

Pool Name	OraDevPool
Pool Type	FBA
Emulation	FBA
Enabled Cap (GB)	2014
Used Cap (GB)	1168

Figure 3. Symmetrix Management Console

Either the command line interface (CLI) or SMC graphic user interface (GUI) can be used to manage Virtual LUN, Virtual Provisioning, and FAST VP operations and it is the user’s responsibility to decide which management tool they prefer. When discussing FAST VP, the focus of the paper will be on SMC as automation tools are commonly managed via a GUI, and it is assumed that manual-initiated data migrations are often called from scripts and therefore VLUN VP examples will be shown using CLI. Other management tools that can help with FAST VP monitoring are Ionix™ ControlCenter® Storage Scope™, Symmetrix Performance Analyzer (SPA), and others. Their use is not covered in this paper.

Symmetrix VMAX Virtual Provisioning

Introduction to Virtual Provisioning

Symmetrix Virtual Provisioning, the Symmetrix implementation of what is commonly known in the industry as “thin provisioning,” enables users to simplify storage management and increase capacity utilization by sharing storage among multiple applications and only allocating storage as needed from a shared “virtual pool” of physical disks.

Symmetrix thin devices are logical devices that can be used in many of the same ways that Symmetrix standard devices have traditionally been used. Unlike traditional Symmetrix devices, thin devices do not need to have physical storage preallocated at the time the device is created and presented to a host (although in many cases customers interested only in wide striping and ease of management choose to fully preallocate the thin devices). A thin device is not usable until it has been bound to a shared storage pool known as a thin pool. Multiple thin devices may be bound to any given thin pool. The thin pool is comprised of devices called data devices that provide the actual physical storage to support the thin device allocations.

When a write is performed to a part of any thin device for which physical storage has not yet been allocated, the Symmetrix allocates physical storage from the thin pool for that portion of the thin device only. The Symmetrix operating environment, Enginuity, satisfies the requirement by providing a block of storage from the thin pool called a thin device extent. This approach reduces the amount of storage that is actually consumed.

The minimum amount of physical storage that can be reserved at a time for the dedicated use of a thin device is referred to as a data device extent. The data device extent is allocated from any one of the data devices in the associated thin pool. Allocations across the data devices are balanced to ensure that an even distribution of allocations occurs from all available data devices in the thin pool (also referred to as wide striping).

For Symmetrix, the thin device extent size is the same as the data device extent size, which is 12 Symmetrix tracks or 768 KB. As a note, there is no reason to match the LVM stripe depth with the thin device extent size. Oracle commonly accesses data either by random single block read/write operations (usually 8 KB in size) or sequentially by reading large portions of data. In either case there is no advantage or disadvantage to match the LVM stripe depth to the thin device extent size as single block read/writes operate on a data portion that is smaller than the LVM stripe depth anyway. For sequential operations, if the data is stored together in adjacent locations on the devices, the read operation will simply continue to read data on each LUN (every time the sequential read wraps to that same LUN) regardless of the stripe depth. If the LVM striping caused the data to be stored randomly on the storage devices then the sequential read operation will turn into a storage random read of large I/Os spread across all the devices.

When a read is performed on a thin device, the data being read is retrieved from the appropriate data device in the thin pool to which the thin device is associated. If for some reason a read is performed against an unallocated portion of the thin device, zeros are returned to the reading process.

When more physical data storage is required to service existing or future thin devices, for example, when a thin pool is approaching full storage allocations, data devices can be added to existing thin pools dynamically without causing a system outage. New thin devices can also be created and bound to an existing thin pool at any time.

When data devices are added to a thin pool they can be in an enabled or disabled state. In order for the data device to be used for thin extent allocation it needs to be in the enabled state. For it to be removed from the thin pool, it needs to be in a disabled state. Symmetrix automatically initiates a drain operation on a disabled data device without any disruption to the application. Once all the allocated extents are drained to other data devices, a data device can be removed from the thin pool.

The following figure depicts the relationships between thin devices and their associated thin pools. Thin Pool A contains six data devices, and thin Pool B contains three data devices. There are nine thin devices associated with thin Pool A and three thin devices associated with thin pool B. The data extents for thin devices are distributed on various data devices as shown in Figure 4.

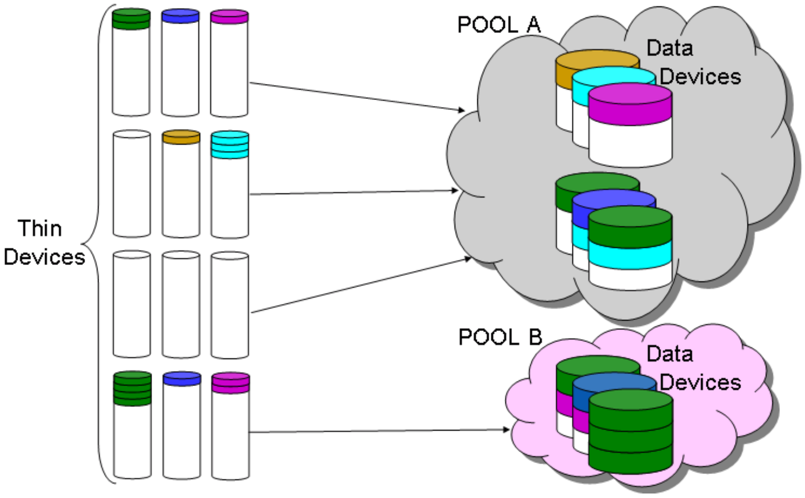


Figure 4. Thin devices and thin pools containing data devices

The way thin extents are allocated across the data devices results in a form of striping in the thin pool. The more data devices in the thin pool (and the associated physical drives behind them), the wider striping will be, creating an even I/O distribution across the thin pool. Wide striping simplifies storage management by reducing the time required for planning and execution of data layout.

Automated pool rebalancing

Starting with the Enginuity 5874 Q4 2009 service release and Solutions Enabler 7.1, automated pool rebalancing allows the user to run a balancing operation that will redistribute data evenly across the enabled data devices in the thin pool. Because the thin extents are allocated from the thin pool in round-robin fashion, the rebalancing mechanism will be used primarily when adding data devices to increase thin pool capacity. If automated pool rebalancing is not used, existing data extents will not benefit from the added data devices as they will not be redistributed.

The balancing algorithm will calculate the minimum, maximum, and mean used capacity values of the data devices in the thin pool. The Symmetrix will then move thin device extents from the data devices with the highest used capacity to those with the lowest until the pool is balanced. Symmetrix VMAX automated pool rebalancing allows nondisruptive extension of a thin pool in increments as needed, maximizing performance and minimizing TCO.

Virtual Provisioning natively offers wide striping and balanced drive access across the enabled data devices in the pool. Automated pool rebalancing allows redistribution of data extents in the thin pool when new devices are made available. The balanced distribution of data extents over a larger set of data devices would result in balanced drive utilization at the Symmetrix back end. This helps achieve higher overall application performance.

Symmetrix Virtual LUN (VLUN) technology

Enginuity 5874 and later provide an enhanced version of Symmetrix Virtual LUN (VLUN) software to enable transparent, nondisruptive data mobility of devices between storage tiers and/or RAID protections as shown in Figure 5. VLUN technology provides users with the ability to move Symmetrix “thick” logical devices between drive types, such as high-performance Enterprise Flash Drives (EFDs), Fibre Channel drives, or high-capacity low-cost SATA drives, and at the same time change their RAID protection.

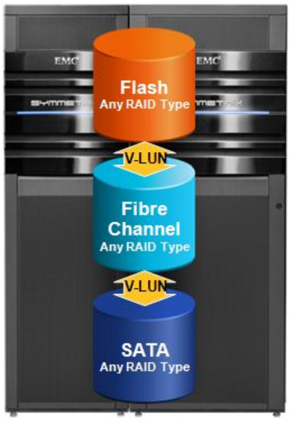


Figure 5. Enhanced Virtual LUN

VLUN migration is independent of host operating systems or applications, and during the migration the devices remain fully accessible to database transactions. While the back-end device characteristics change (RAID protection and/or physical drive type) the identities of the devices being migrated remain the same to the host, allowing seamless online migration. VLUN migration is fully integrated with Symmetrix replication technology and maintains consistency of source/target device relationships in replications such as SRDF, TimeFinder/Clone, TimeFinder/Snap, or Open Replicator.

The advantages of migrating data using storage technology are ease of use, efficiency, and simplicity. Data is migrated in the Symmetrix back end without needing any SAN or host resources, which increases migration efficiency. The migration is a safe operation as the target device is treated internally as just another "mirror" of the logical device, although with its own RAID protection and drive type. At the end of the migration the data on the original "mirror" is formatted to preserve security. Finally, since the identity of source devices doesn't change, moving between storage tiers is easy and doesn't require additional host change control, backup script updates, changes in filesystem mount points, volume manager, or others. The migration pace can be controlled using Symmetrix quality of service (symqos) commands.

VLUN migration helps customers to implement an Information Lifecycle Management (ILM) strategy for their databases, such as the move of the entire database, tablespaces, partitions, or ASM disk groups between storage tiers. It also allows adjustments in service levels and performance requirements to application data. For example, often application storage is provisioned before clear performance requirements are known. At a later time, once the requirements are better understood, it is easy to make any adjustment to increase user experience and ROI using the correct storage type.

Figure 6 shows an example of performing a Virtual LUN migration of an ASM disk group "+Sales" with 20 x 50 GB logical devices (ASM members). The migration source devices are spread across 40 x 300 GB drives and protected with RAID 1. The migration target devices are spread across only 4 x 400 GB EFDs and protected with RAID 5.

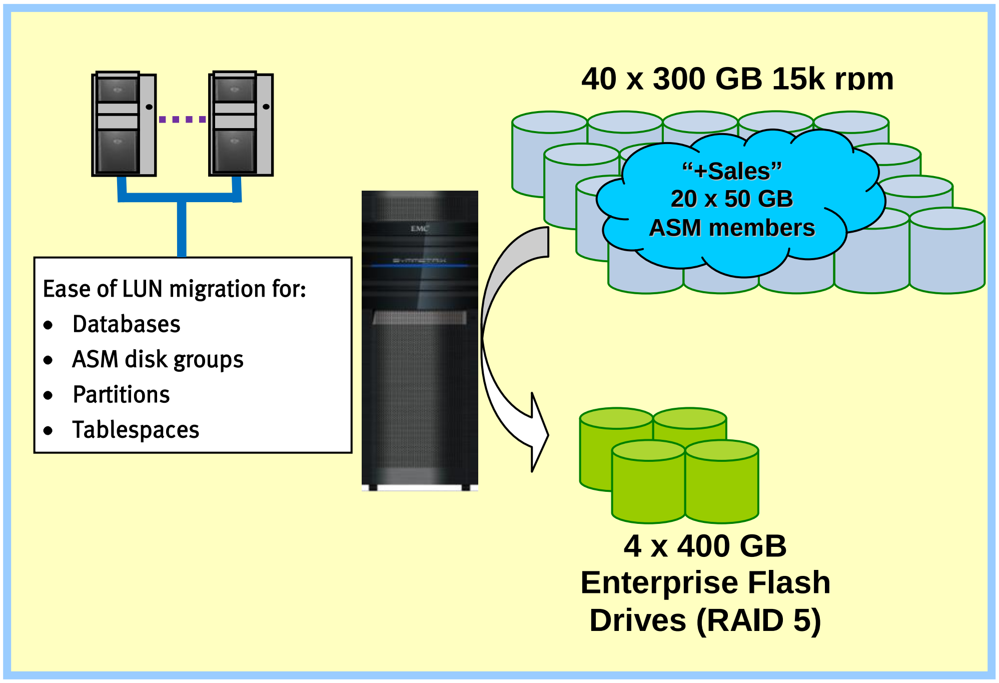


Figure 6. Migration example using VLUN technology

The following steps demonstrate the use of VLUN, based on the example in Figure 6.

1. **Optional: Verify information for a migration session called *Sales_mig***

   ```
   symmigrate -name Sales_mig –file Sales_ASM.txt –sid <Symm ID> validate
   ```

   The file Sales_ASM.txt contains the list of source and target migration devices:

   ```
   0100 0C00
   ...  ...
   0113 0C13
   ```

2. **Perform the migration**

   ```
   symmigrate -name Sales_mig –file Sales_ASM.txt –sid <Symm ID> establish
   ```

3. **Follow the migration progress and rate at 60-second intervals**

   ```
   symmigrate -name Sales_mig –file Sales_ASM.txt –sid <Symm ID> query –i 60
   ```

4. **Terminate the migration session after completion**

   ```
   symmigrate -name Sales_mig –file Sales_ASM.txt –sid <Symm ID> terminate
   ```

5. **Optional: Control the migration pace**

   Create a Symmetrix DG with the source devices

   ```
   symdg create Sales_dg
   symld –g Sales_dg –range 0100:0113 addall
   ```

   Control the copy pace using the DG

   ```
   symqos –g Sales_dg set MIR pace 8
   ```

Symmetrix Virtual LUN VP (VLUN VP) Mobility technology

Introduced in Enginuity 5875, EMC Symmetrix VMAX VLUN VP enables transparent, nondisruptive data mobility of thin devices between storage tiers and/or RAID protections. VLUN VP benefits and usage are almost identical to VLUN with the exception that while VLUN operated on "thick" devices, VLUN VP operates only on thin devices, and migrates only the allocated extents of a thin device to a single target thin pool. As a result, at the end of the migration the thin device will share the storage tier and RAID protection of the target thin pool.

Note that when using VLUN VP on devices under FAST VP control, it is recommended to *pin* the thin devices to the target thin pool so FAST VP won't move them to other tiers until the user is ready. When thin devices under FAST VP control are pinned to a thin pool, FAST VP continues to collect their statistics, but it won't issue move plans for them.

VLUN VP enables customers to move Symmetrix thin devices without disrupting user applications and with minimal impact to host I/O. Users may move thin devices between thin pools to:

- Change the drive media on which the thin devices are stored
- Change the thin device RAID protection level
- Move a thin device that was managed by FAST VP (and may be spread across multiple tiers, or thin pools) to a single thin pool

While VLUN VP has the ability to move all allocated thin device extents from one pool to another, it also has the ability to move specific thin device extents from one pool to another, and it is this feature that is the basis for FAST VP.

Symmetrix FAST

Introduced in the Enginuity 5874 Q4 service release, EMC Symmetrix VMAX FAST is Symmetrix software that utilizes intelligent algorithms to continuously analyze device I/O activity and generate plans for moving and swapping devices for the purposes of allocating or re-allocating application data across different storage tiers within a Symmetrix array. FAST proactively monitors workloads at the Symmetrix device (LUN) level in order to identify "busy" devices that would benefit from being moved to higher-performing drives such as EFD. FAST will also identify less "busy" devices that could be relocated to higher-capacity, more cost-effective storage such as SATA drives without altering performance.

Time windows can be defined to specify when FAST should collect performance statistics (upon which the analysis to determine the appropriate storage tier for a device is based), and when FAST should perform the configuration changes necessary to move devices between storage types. Movement is based on user-defined storage tiers and FAST policies.

The primary benefits of FAST include:

- Eliminating manually tiering applications when performance objectives change over time
- Automating the process of identifying volumes that can benefit from EFD or that can be kept on higher-capacity, less-expensive SATA drives without impacting performance
- Improving application performance at the same cost, or providing the same application performance at lower cost. Cost is defined as acquisition (both hardware and software), space/energy, and management expense
- Optimizing and prioritizing business applications, allowing customers to dynamically allocate resources within a single array
- Delivering greater flexibility in meeting different price/performance ratios throughout the lifecycle of the information stored

Symmetrix FAST VP

FAST VP was introduced in Enginuity 5875. FAST VP shares the same benefits as FAST with the difference that while FAST operates on a full thick device level, FAST VP operates at a sub-LUN level and is based on Virtual Provisioning. FAST VP automates the identification of thin device extents for the purposes of re-allocating application data across different performance tiers within a single array. FAST VP proactively monitors workloads at a sub-LUN level in order to identify active areas that would benefit from being moved to higher-performing drives. FAST VP will also identify less active sub-LUN areas that could be moved to higher-capacity drives, without existing performance being affected.

LUN and sub-LUN access skewing

As described earlier, almost any application causes access skewing at a LUN or sub-LUN level. In other words, some portions of the data are heavily accessed, some are accessed to a lesser degree, and often some portions are hardly accessed at all. Because DBAs tend to plan for the worst-case peak workloads they commonly place almost all data into a single storage tier based on fast FC drives (10k or 15k rpm). Based on the availability of multiple storage tiers and FAST VP technology, a more efficient storage tiering strategy can be deployed that will place the correct data on the right storage tier for it.

FAST VP and Virtual Provisioning

FAST VP is based on Virtual Provisioning technology. Virtual Provisioning as explained earlier allows the creation and use of virtual devices (commonly referred to as thin devices) that are host-addressable, cache-only pointer-based devices. Once the host starts using the thin devices, their data is allocated in commonly shared pools called thin pools. A thin pool is simply a collection of Symmetrix regular devices of the same drive technology and RAID protection (for example, 50 x 100 GB RAID 5 15k rpm FC devices can be grouped into a thin pool called FC15k_RAID5). Because the thin pool devices store the pointer-based thin devices' data, they are also referred to as data devices. Data in the thin pool is always striped, taking advantage of all the physical drives behind the thin pool data devices. This allows both improved performance as well as ease of deployment and storage provisioning. In addition, as data devices are added or removed from the thin pool, their data will be rebalanced (restriped) seamlessly as well. In short, Virtual Provisioning has many deployment advantages in addition to being the base technology for FAST VP.

One can start understanding how FAST VP benefits from this structure. Since the thin device is pointer-based, and its actual data is stored in thin pools based on distinct drive type technology, when FAST VP moves data between storage tiers it simply migrates the data between the different thin pools and updates the thin device pointers accordingly. To the host, the migration is seamless as the thin device maintains the exact same LUN identity. At the Symmetrix storage, however, the data is migrated between thin pools without any application downtime.

FAST VP elements

FAST VP has three main elements — storage tiers, storage groups, and FAST policies — as shown in Figure 7.

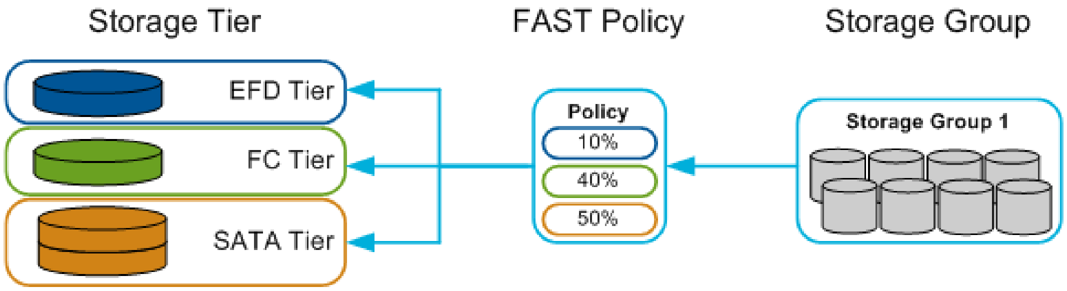


Figure 7. FAST managed objects

- Storage tiers are the combination of drive technology and RAID protection available in the VMAX array. Examples for storage tiers are RAID 5 EFD, RAID 1 FC, RAID 6 SATA, and so on. Since FAST VP is based on Virtual Provisioning, the storage tiers for FAST VP contain one to three thin pools of the same drive type and RAID protection.
- Storage groups are collections of Symmetrix host-addressable devices. For example, all the devices provided to an Oracle database can be grouped into a storage group. While a storage group can contain both thin and thick devices, FAST VP will operate only on the thin devices in a given storage group.
- A FAST VP policy combines storage groups with storage tiers, and defines the configured capacities, as a percentage, that a storage group is allowed to consume on that tier. For example a FAST VP policy can define 10 percent of its allocation to be placed on EFD_RAID 5, 40 percent on FC15k_RAID 1, and 50 percent on SATA_RAID 6 as shown in Figure 7. Note that these allocations are the maximum allowed. For example, a policy of 100 percent on each of the storage tiers means that FAST VP has liberty to place up to 100 percent of the storage group data on any of the tiers. When combined, the policy must total at least 100

percent, but may be greater than 100 percent as shown in Figure 8. In addition the FAST VP policy defines exact time windows for performance analysis, data movement, data relocation rate, and other related settings.

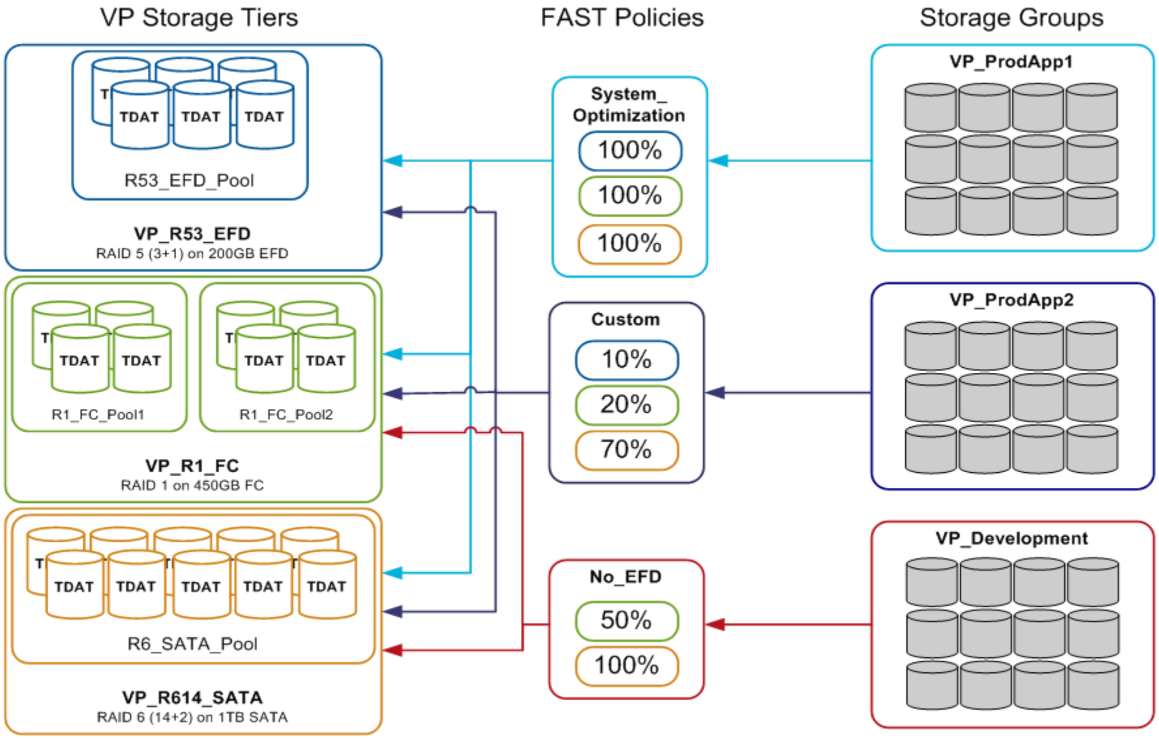


Figure 8. FAST policy association

FAST VP operates in the storage array based on the policy allocation limits for each tier ("Compliance"), and in response to the application workload ("Performance"). During the Performance Time Window FAST will gather performance statistics for the controlled storage groups. During the Move Time Window FAST will then create move plans (every 10 minutes) that will accommodate any necessary changes in performance or due to compliance changes. Therefore FAST VP operates in reactions to changes in workload or capacities, in accordance to the policy.

FAST VP Performance Time Window considerations

There is no one Performance Time Window recommendation that is generically applicable to all customer environments. Each site will need to make the decision based on their particular requirements and SLAs. Collecting statistics 24x7 is simple and the most comprehensive approach; however, overnight and daytime I/O profiles may differ greatly, and evening performance may not be as important as daytime performance. This difference can be addressed by simply setting the collection policy to be active only during the daytime from 7 A.M. to 7 P.M., Monday to Friday. This policy is best suited for applications that have consistent I/O loads during traditional business hours. Another approach would be to only collect statistics during peak times on specific days. This is most beneficial to customers whose I/O profile has very specific busy periods, such as the A.M. hours of Mondays. By selecting only the peak hours for statistical collection the site can ensure that the data that is most active during peak periods gets the highest priority to move to a high-performance tier. The default Performance Time Window is set for 24x7 as the norm but can be easily changed using CLI or SMC.

FAST VP Move Time Window considerations

Choosing a FAST VP Move Time Window allows a site to make a decision about how quickly FAST VP responds to changes in the workload. Allowing it to move data at any time of the day lets FAST VP quickly adapt to changing I/O profiles but may add activity to the Symmetrix back end during these peak times. Alternatively, the FAST VP Move Time Window can be set to specific lower activity hours to prevent FAST activity from interfering with online activity. One such case would be when FAST is initially implemented on the array when the amount of data being moved could be substantial. In either case FAST VP would attempt to make the move operations as efficiently as possible by only moving allocated extents, and with sub-LUN granularity the move operations are focused on just the data sets that need to be promoted or demoted.

The FAST VP Relocation Rate (FRR) is a quality-of-service setting for FAST VP and affects the “aggressiveness” of data movement requests generated by FAST VP. FRR can be set between 1 and 10, with 1 being the most aggressive, to allow the FAST VP migrations to complete as fast as possible, and 10 being the least aggressive. With the release of FAST VP and Enginuity 5875, the default FRR is set to 5 and can be easily changed dynamically. An FRR of 6 was chosen for the use cases in this paper.

FAST VP architecture

There are two components of FAST VP: Symmetrix microcode and the FAST controller.

The Symmetrix microcode is a part of the Enginuity storage operating environment that controls components within the array. The FAST controller is a service that runs on the Symmetrix service processor.

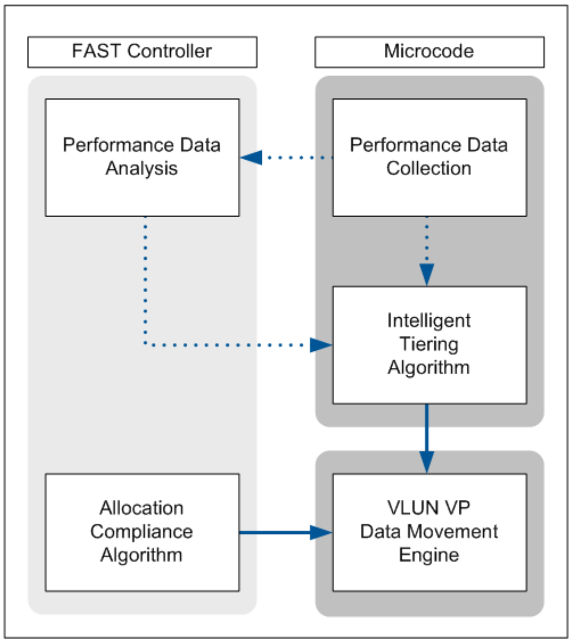

FAST Controller
Microcode
Performance Data Analysis
Performance Data Collection
Intelligent Tiering Algorithm
Allocation Compliance Algorithm
VLUN VP Data Movement Engine

Figure 9. FAST VP components

When FAST VP is active, both components participate in the execution of two algorithms to determine appropriate data placement:

- Intelligent tiering algorithm

  The intelligent tiering algorithm uses performance data collected by the microcode, as well as supporting calculations performed by the FAST controller, to issue data movement requests to the VLUN VP data movement engine.

- Allocation compliance

  The allocation compliance algorithm enforces the upper limits of storage capacity that can be used in each tier by a given storage group by also issuing data movement requests to the VLUN VP data movement engine.

Data movements performed by the microcode are achieved by moving allocated extents between tiers. The size of data movement can be as small as 768 KB, representing a single allocated thin device extent, but will more typically be an entire extent group, which is 10 thin device extents, or 7.5 MB.

FAST VP has two modes of operation, Automatic or Off. When operating in Automatic mode, data analysis and data movements will occur continuously during the defined windows. In Off mode, performance statistics will continue to be collected, but no data analysis or data movements will take place.

Virtual Provisioning and Oracle databases

Strategies for thin pool allocation with Oracle databases

Oracle Database file initialization

Using Virtual Provisioning in conjunction with Oracle databases provides the benefits mentioned earlier, such as reducing future server impact during LUN provisioning, increasing storage utilization, native striping in the thin pool, and ease and speed of creating and working with thin devices. However, as commonly known, when Oracle initializes new files, such as log, data and temp files, it fully allocates the file space by writing non-zero information (metadata) to each initialized block. This will cause the thin pool to allocate the amount of space that is being initialized by the database. As database files are added, more space will be allocated in the pool. Due to Oracle file initialization, and in order to get the most benefit from a Virtual Provisioning infrastructure, a strategy for sizing files, pools, and devices should be developed in accordance to application and storage management needs. Some strategy options are explained next.

Oversubscription

An oversubscription strategy is based on using thin devices with a total capacity greater than the physical storage in the pool(s) they are bound to. This can increase capacity utilization by sharing storage among applications, thereby reducing the amount of allocated but unused space. The thin devices each appear to be a full-size device to the application, while in fact the thin pool can't accommodate the total thin LUN capacity. Since Oracle database files initialize their space even though they are still empty, it is recommended that instead of creating very large data files that remain largely empty for most of their lifetime, smaller data files should be considered to accommodate near-term data growth. As they fill up over time, their size can be increased, or more data files added, in conjunction with the capacity increase of the thin pool. The Oracle auto-extend feature can be used for simplicity of management, or DBAs may prefer to use manual file size management or addition.

An oversubscription strategy is recommended for database environments when database growth is controlled, and thin pools can be actively monitored and their size increased when necessary in a timely manner.

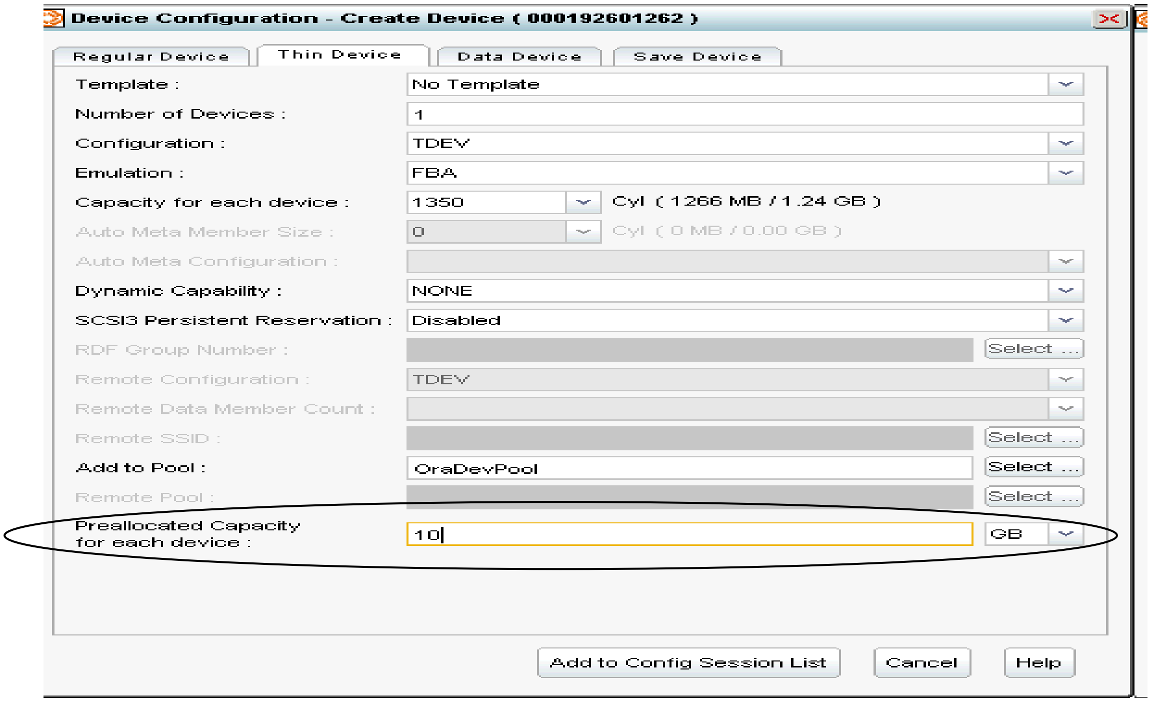
Undersubscription

An undersubscription strategy is based on using thin devices with a total capacity smaller than the physical storage in the pool(s) they are bound to. This approach doesn't necessarily improve storage capacity utilization but still makes use of wide striping, thin pool sharing, and other benefits of Virtual Provisioning. In this case the data files can be sized to make immediate use of the full thin device size, or alternatively, auto-extend or manual file management can be used.

Undersubscribing is recommended when data growth is unpredictable, when multiple small databases share a large thin pool to benefit from wide striping, or when an oversubscriptioned environment is considered unacceptable.

Thin device preallocation

A third option exists that can be used with either oversubscription or undersubscription, and has become very popular for Oracle databases. When the DBAs like to guarantee that space is reserved for the databases' thin devices, they can use thin device preallocation. While this reduces potential capacity utilization benefits for the thin pool, it still enables users to achieve easier data layout with wide striping. A thin device can preallocate space in the pool, even before data was written to it. Figure 10 shows an example of creating 10 x 29.30 GB thin devices, and preallocating 10 GB in the pool for each of them. The example shows an SMC screen (a similar operation can be done using the Symmetrix CLI). When preallocation is used Oracle database customers often preallocate the whole thin device (reducing the storage capacity optimization benefits). In effect each thin device therefore fully claims its space in the thin pool, eliminating a possible thin pool out-of-space condition. It is also possible to preallocate a portion of the thin device (like the 10 GB in the example) to match the size of the application file. For example, ASM disks can be set smaller than their actual full size, and later be resized dynamically without any disruption to the database application. In this case an ASM disk group can be created from these 10 thin devices, only using 10 GB of each disk. At a later time, additional storage on the thin device can be preallocated, and ASM disks resized to match it.



Template : No Template
Number of Devices : 1
Configuration : TDEV
Emulation : FBA
Capacity for each device : 1350 Cyl (1266 MB / 1.24 GB)
Auto Meta Member Size : 0 Cyl (0 MB / 0.00 GB)
Auto Meta Configuration :
Dynamic Capability : NONE
SCSI3 Persistent Reservation : Disabled
RDF Group Number :
Remote Configuration : TDEV
Remote Data Member Count :
Remote SSID :
Add to Pool : OraDevPool
Remote Pool :
Preallocated Capacity for each device : 10 GB

Figure 10. Symmetrix Management Console and thin device preallocation example

Planning thin pools for Oracle databases

Planning thin pools for Oracle environments requires some attention to detail but the advantage of using thin pools is that the environment is flexible. By using thin devices, performance of the database can be improved over thick devices because thin devices are striped evenly over all the physical drives in the pool. For typical OLTP Oracle databases this provides the maximum number of physical devices to service the workload. If a database starts on a pool of, say, 64 physical devices, and the load to those devices is too heavy, the pool can be expanded dynamically without interruption to the application, to spread the load over more physical drives.

In general thin pools should be configured to meet at least the initial capacity requirements of all applications that will reside in the pool. The pool should also contain enough physical drives to service the expected back-end physical drive workload. Customers can work with their local EMC account team for recommendations on how to size the number of physical drives.

For RAID protection, thin pools are no different in terms of reliability and physical drive performance than existing drives today. If an application is deployed on RAID 5 (3+1) today, there is no reason to change the protection for thin pools. Likewise if an application is deployed on RAID 1 or RAID 5 (7+1), then the thin pool should be configured to match. Both RAID 1 and RAID 5 protect from a single-drive failure, and RAID 6 protects from two-drive failures. A RAID 1 group resides on two physical drives; a RAID 5 (3+1) group resides on four physical drives, and so on. When a thin pool is created, it is always created out of similarly configured RAID groups. For example, if we create eight RAID 5 (3+1) data devices and put them into one pool, the pool has eight RAID 5 devices of four drives each. If one of the drives in this pool fails, you are not losing one drive from a pool of 32 drives; rather, you are losing one drive from one of the eight RAID-protected data devices and that RAID group can continue to service read and write requests, in degraded mode, without data loss. Also, as with any RAID group, with a failed drive Enginuity will immediately invoke a hot sparing operation to restore the RAID group to its normal state. While this RAID group is rebuilding, any of the other RAID groups in the thin pool can have a drive failure and there is still no loss of data. In this example, with eight RAID groups in the pool there can be one failed drive in each RAID group in the pool without data loss. In this manner data stored in the thin pool is no more vulnerable to data loss than any other data stored on similarly configured RAID devices. Therefore a protection of RAID 1 or RAID 5 for thin pools is acceptable for most applications and RAID 6 is only required when in situations where additional parity protection is warranted.

The number of thin pools is affected by a few factors. The first is the choice of drive type and RAID protection. Each thin pool is a group of data devices sharing the same drive type and RAID protection. For example, a thin pool that consists of multiple RAID 5 protected data devices based on 15k rpm FC disk can host the Oracle data files for a good choice of capacity/performance optimization. However, very often the redo logs that take relatively small capacity are best protected using RAID 1 and therefore another thin pool containing RAID 1 protected data devices can be used. In order to ensure sufficient spindles behind the redo logs the same set of physical drives that is used for the RAID 5 pool can also be used for the RAID 1 thin pool. Such sharing at the physical drive level, but separation at the thin pool level, allows efficient use of drive capacity without compromising on the RAID protection choice. Oracle Fast Recovery Area (FRA)[1], for example, can be placed in a RAID 6 protected SATA drive's thin pool.

Therefore the choice of the appropriate drive technology and RAID protection is the first factor in determining the number of thin pools. The other factor has to do with the business owners. When applications share thin pools they are bound to the same set of data devices and spindles, and they share the same overall thin pool capacity and performance. If business owners require their own control over thin pool management they will likely need a separate set of thin pools based on their needs. In general, however, for ease of manageability it is best to keep the overall number of thin pools low, and allow them to be spread widely across many drives for best performance.

Planning thin devices for Oracle databases

Thin device LUN sizing

The maximum size of a standard thin device in a Symmetrix VMAX is 240 GB. If a larger size is needed, then a metavolume comprised of thin devices can be created. When host striping is used, like Oracle ASM, it is recommended that the metavolume be concatenated rather than striped since the host will provide a layer of striping, and the thin pool is already striped based on data device extents. Concatenated metavolumes also support fast expansion capabilities, as new metavolume members can be easily appended to the existing concatenated metavolume. This functionality may be applicable when the provisioned thin device has become fully allocated at the host level, and it is required to further increase the thin device to gain additional space. Note that it is not recommended to provision applications with a low number of very large LUNs. The reason is that each LUN provides the host with an additional I/O queue to which the host operating system can stream I/O requests and parallelize the workload. Host software and HBA drivers tend to limit the amount of I/Os that can be queued at a time to a LUN and therefore to avoid host queuing bottlenecks under heavy workloads, it is better to provide the application with multiple, smaller LUNs rather than very few and large LUNs.

Striped metavolumes are supported with Virtual Provisioning and there may be workloads that will benefit from multiple levels of striping (for example, for Oracle redo logs when SRDF/S is used, and host striping is not available).

When oversubscription is used, the thin pool can be sized for *near-term* database capacity growth, and the thin devices for *long-term* LUN capacity needs. Since the thin LUNs don't take space in the pool until data is written to them[2], this method optimizes storage capacity utilization and reduces the database and application impact as they continue to grow. Note, however, that the larger the device the more

[1] Flash Recovery Area was renamed by Oracle to Fast Recovery Area in database release 11gR2.
[2] When thin device preallocation is not used

metadata is associated with it and tracked in the Symmetrix cache. Therefore the sizing should be reasonable and realistic to limit unnecessary cache overhead, as small as it is.

Thin devices and ASM disk group planning

Thin devices are presented to the host as SCSI LUNs. Oracle recommends creating at least a single partition on each LUN to identify the device as being used. On x86-based platforms it is important to align the LUN partition, for example by using fdisk or parted on Linux. With fdisk, after the new partition is created type "x" to enter Expert mode, then use the "b" option to move the beginning of the partition. Either 128 blocks (64 KB) offset or 2,048 blocks (1 MB) offset are good choices and align with the Symmetrix 64 KB cache track size. After assigning Oracle permissions to the partition it can become an ASM disk group member or used in other ways for the Oracle database.

Oracle recommends when using Oracle Automatic Storage Management (ASM) to use a minimum number of ASM disk groups for ease of management. Indeed when multiple smaller databases share the same performance and availability requirements they can also share ASM disk groups; however, larger, more critical databases may require their own ASM disk groups for better control and isolation. EMC best practice for mission-critical Oracle databases is to create a few ASM disk groups based on the following guidelines:

- +GRID: Starting with database 11gR2 Oracle has merged Cluster Ready Services (CRS) and ASM and they are installed together as part of Grid installation. Therefore when the clusterware is installed the first ASM disk group is also created to host the quorum and cluster configuration devices. Since these devices contain local environment information such as hostnames and subnet masks, there is no reason to clone or replicate them. EMC best practice starting with Oracle Database 11.2 is to only create a very small disk group during Grid installation for the sake of CRS devices and not place any database components in it. When other ASM disk groups containing database data are replicated with storage technology they can simply be mounted to a different +GRID disk group at the target host or site, already with Oracle CRS installed with all the local information relevant to that host and site. Note that while external redundancy (RAID protection is handled by the storage array) is recommended for all other ASM disk groups, EMC recommends high redundancy *only* for the +GRID disk group. The reason is that Oracle automates the number of quorum devices based on redundancy level and it will allow the creation of more quorum devices. Since the capacity requirements of the +GRID ASM disk group are tiny, very small devices can be provisioned (High redundancy implies three copies/mirrors and therefore a minimum of three devices is required).
- +DATA, +LOG: While separating data and log files to two different ASM disk groups is optional, EMC recommends it in the following cases:
    - When TimeFinder is used to create a clone (or snap) that is a valid backup image of the database. The TimeFinder clone image can serve as a source

    for RMAN backup to tape, and/or be opened for reporting (read-only), and so on. However the importance of such a clone image is that it is a valid full backup image of the database. If the database requires media recovery, restoring the TimeFinder clone back to production takes only seconds – regardless of the database size! This is a huge saving in RTO and in a matter of a few seconds archive logs can start being applied as part of media recovery roll forward. When such a clone doesn't exist the initial backup set has to be first restored from tape/VTL prior to applying any archive log, which can add a significant amount of time to recovery operations. Therefore, when TimeFinder is used to create a backup image of the database, in order for the restore to not overwrite the online logs, they should be placed in separate devices and a separate ASM disk group

  - Another reason for separation of data from log files is performance and availability. Redo log writes are synchronous and require to complete in the least amount of time. By having them placed in separate storage devices the commit writes won't have to share the LUN I/O queue with large async buffer cache checkpoint I/Os. Having the logs in their own devices makes it available to use one RAID protection for data files (such as RAID 5), and another for the logs (such as RAID 1).

- +TEMP: When storage replication technology is used for disaster recovery, like SRDF/S, it is possible to save bandwidth by not replicating temp files. Since temp files are not part of a recovery operation and quick to add, having them on separate devices allows bandwidth saving, but adds to the operations of bringing up the database after failover. While it is not required to separate temp files it is an option and the DBA may choose to do it anyway for performance isolation reasons if that is their best practice.

- +FRA: Fast Recovery Area typically hosts the archive logs and sometimes flashback logs and backup sets. Since the I/O operations to FRA are typically sequential writes, it is usually sufficient to have it located on a lower tier such as SATA drives. It is also an Oracle recommendation to have FRA as a separate disk group from the rest of the database to avoid keeping the database files and archive logs or backup sets (that protect them) together.

Thin pool reclamation with the ASM Reclamation Utility (ASRU)

In general, Oracle ASM reuses free/deleted space under the high watermark very efficiently. However, when a large amount of space is released, for example after the deletion of a large tablespace or database, and the space is not anticipated to be needed soon by that ASM disk group, it is beneficial to free up that space in both the disk group and thin pool.

To simplify the storage reclamation of thin pool space no longer needed by ASM objects, Oracle and storage partners have developed the ASM Storage Reclamation Utility. ASRU in conjunction with Symmetrix Space Reclamation helps in consolidating the Oracle ASM disk group, and reclamation of the space that was freed in the ASM disk group, from the Symmetrix storage array. The integration of Symmetrix with ASRU is covered in the white paper Implementing Virtual Provisioning on EMC Symmetrix VMAX with Oracle Database 10g and 11g.

FAST VP and Oracle databases

FAST VP integrates very well with Oracle databases. As explained earlier, applications tend to drive most of the workload to a subset of the database, and very often, just a small subset of the whole database. That subset is a candidate for performance improvement and therefore uptiering by FAST VP. Other database subsets can either remain where they are or be down-tiered if they are mostly idle (for example, unused space or historic data maintained due to regulations). If we look at Oracle ASM, it natively stripes the data across its members, spreading the workload across all storage devices in the ASM disk group. From the host it may look as if all the LUNs are very active but in fact, in almost all cases just a small portion of each LUN is very active. Figure 11 shows an example of I/O read activity, as experienced by the Symmetrix storage array, to a set of 15 ASM devices (X-axis) relative to the location on the devices (Y-axis). The color reflects I/O activity to each logical block address on the LUN (LBA), where blue indicates low activity and red high. It is easy to see in this example that while ASM stripes the data and spreads the workload evenly across the devices, not all areas on each LUN are “hot,” and FAST VP can focus on the hot areas alone and uptier them. It can also down-tier the idle areas (or leave them in place, based on the policy allocations). The result will be improved performance, cost, and storage efficiency.

Even if ASM is not in use other volume managers tend to stripe the data across multiple devices and will therefore benefit from FAST VP in a similar way. When filesystems alone are used we can look at a sub-LUN skewing inside the filesystem rather than a set of devices. The filesystem will traditionally host multiple data files, each containing database objects in which some will tend to be more active than others as discussed earlier, creating I/O access skewing at a sub-LUN level.

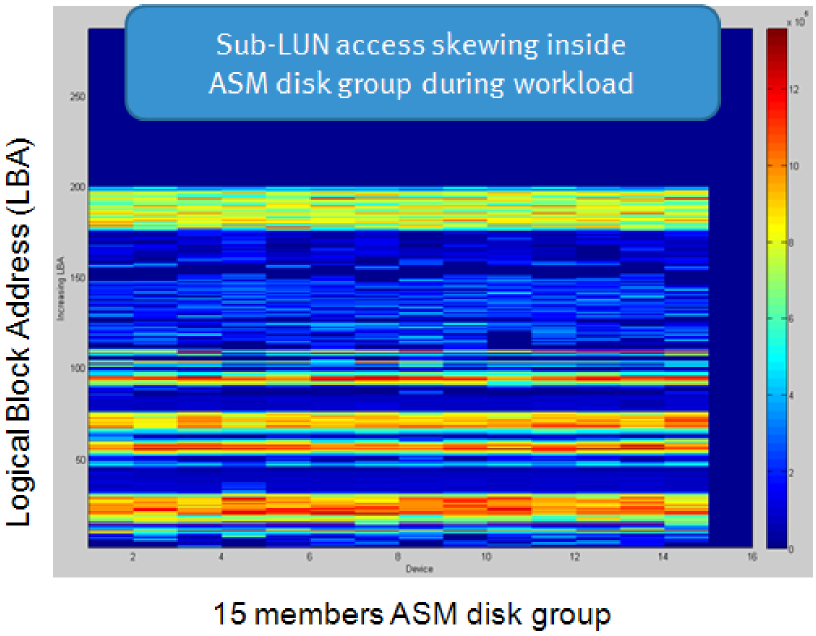


Figure 11. “Heat” map of ASM disks showing sub-LUN skewing

At the same time there are certain considerations that need to be understood in relationship to FAST VP and planned for. One of them is instantaneous changes in workload characteristics and the other is changes in data placement initiated by the host such as ASM rebalance.

Instantaneous changes in workload characteristics

Instantaneous changes in workload characteristics, such as quarter-end or year-end reports, may put a heavy workload on portions of the database that are not accessed daily and may have been migrated to a lower-performance tier. Symmetrix is optimized to take advantage of very large cache (up to 1 TB raw) and has efficient algorithms to prefetch data and optimize disk I/O access. Therefore Symmetrix VMAX will handle most workload changes effectively and no action needs to be taken by the user. On the other hand the user can also assist by modifying the FAST VP policy ahead of such activity when it is known and expected, and by changing the Symmetrix priority controls and cache partitioning quotas if used. Since such events are usually short term and only touch each data set once it is unlikely (and not desirable) for FAST VP to migrate data at that same time and it is best to simply let the storage handle the workload appropriately. If the event is expected to last a longer period of time (such as hours or days), then FAST VP, being a reactive mechanism, will actively optimize the storage allocation as it does natively.

Changes in data placement initiated by the host (such as ASM rebalance)

Changes in data placement initiated by the host can be due to filesystem defrag, volume manager restriping, or even simply a user moving database objects. When Oracle ASM is used the data is automatically striped across the disk group. There are certain operations that will cause ASM to restripe (rebalance) the data, effectively moving existing allocated ASM extents to a new location, which may cause the storage tiering optimized by FAST VP to temporarily degrade until FAST VP re-optimizes the database layout. ASM rebalance commonly takes place when devices are added or dropped from the ASM disk group. These operations are normally known in advance (although not always) and will take place during maintenance or low-activity times. Typically new thin devices given to the database (and ASM) will be bound to a medium- or high-performance storage tier, such as FC or EFD. Therefore when such devices are added, ASM will rebalance extents into them, and it is unlikely that database performance will degrade much afterward (since they are already on a relatively fast storage tier). If such activity takes place during low-activity or maintenance time it may be beneficial to disable FAST VP movement until it is complete and then let FAST VP initiate a move plan based on the new layout. FAST VP will respond to the changes and re-optimize the data layout. Of course it is important that any new devices that are added to ASM should be also added to the FAST VP controlled storage groups so FAST VP can operate on them together with the rest of the database devices.

Which Oracle objects to place under FAST VP control

Very often storage technology is managed by a different group from the database management team and coordination is based on need. In these cases when devices are provisioned to the database they can be placed under FAST VP control by the storage team without clear knowledge on how the database team will be using them. Since FAST VP analyzes the actual I/O workload based on the FAST policy it will actively optimize the storage tiering of all controlled devices.

However, when more coordination takes place between the database and storage administrators it might be best to focus the FAST VP optimization on database data files, and leave other database objects such as logs and temp space outside of FAST VP control. The reason is that redo logs, archive logs, and temp space devices experience sequential read and write activity. All writes in Symmetrix go to cache and are acknowledged immediately to the host (regardless of storage tier). For sequential reads, the different disk technologies at the storage array will have minimal impact due to I/O prefetch and reduced disk head movement (in contrast to random read activity).

FAST VP algorithms place higher emphasis on improving random read I/O activity although they also take into consideration writes and sequential reads activity. Placing only data files under FAST VP control will reduce the potential competition over the EFD tier by database objects that may have a high I/O load but are of less importance to consume precious capacity on that tier. However, as mentioned earlier, when all database devices are under FAST VP control, such objects may uptier, but with a lesser priority than objects with random read activity (such as data files with a typical I/O profile).

A different use case for FAST VP usage could be to optimize the storage tiering of sequential read/write devices (like temp files, archive logs) in a separate storage group and FAST VP policy with only SATA and FC tiers included in the FAST VP policy.

In that way the goal is again to eliminate competition over EFD, while allowing dynamic cost/performance optimization for archive logs and temp files between SATA and FC tiers (redo logs are best served by the FC tier in almost all cases).

OLTP vs. DSS workloads and FAST VP

As explained in the previous section, FAST VP places higher emphasis on uptiering a random read workload, although it will try to improve performance of other devices with high I/O activity such as sequential reads and writes. For that reason the active data set of the OLTP applications will have a higher priority to be uptiered by FAST VP over DSS. However, DSS applications can benefit from FAST VP as well. First, data warehouse/BI systems often have large indexes that generate random read activity. These indexes generate an I/O workload that can highly benefit by being uptiered to EFD. Master Data Management (MDM) tables are another example of objects that can highly benefit from the EFD tier. FAST VP also downtiers inactive data. This is especially important in DSS databases that tend to be very large. FAST VP can reduce costs by downtiering the aged data and partitions, and keep the active data set in faster tiers. FAST VP does the storage tiering automatically without having to continuously perform complex ILM actions at the database or application tiers.

Use case examples of Oracle Database 11g and FAST VP

This section covers examples of using Oracle Database 11g with FAST VP. The three use cases are:

1. FAST VP optimization of a single Oracle Database OLTP workload: This use case demonstrates the basic work of FAST VP and how it optimizes the storage allocation of a single Oracle Database from the initial FC tier to all three tiers—SATA, FC, and EFD.
2. FAST VP optimization of two databases sharing an ASM disk group: This use case demonstrates FAST VP optimization when multiple Oracle databases with different workloads are sharing the same ASM disk groups, storage devices, and FAST VP policy.
3. FAST VP optimization of two databases with separate ASM disk groups: This use case demonstrates FAST VP optimization when each database requires its own FAST VP policy for better isolation and control of resources.

Test environment

This section describes the hardware, software, and database configuration used for Oracle databases and FAST VP test cases as seen in Table 1.

Table 1. Test environment

Configuration aspect	Description
Storage array	Symmetrix VMAX
Enginuity	5875
Oracle	CRS and database version 11gR2
EFD	8 x 400 GB EFD
FC	40 x FC 15k rpm 300 GB drives
SATA	32 x SATA 7,200 rpm 1 TB drives
Linux	Oracle Enterprise Linux 5.3
Multipathing	EMC PowerPath® 5.3 SP1
Host	Dell R900

Test Case 1: FAST VP optimization of a single Oracle Database OLTP workload

This section shows an example of the benefits of FAST VP storage tiering optimization of a single Oracle ASM-based database executing an OLTP workload. It highlights the changes in the tier allocation and performance between the beginning and the end for the run. The +DATA ASM disk group starts all on the FC tier and FAST VP migrates idle portions to SATA, and highly active portions to EFD. At the end of the run we can see improved transaction rates and response times and very efficient usage of the three tiers.

The test configuration had two Oracle databases — FINDB (Financial) and HRDB (Human Resource) — sharing ASM disk groups and therefore also a Virtual Provisioning storage group and FAST VP policy, as shown in Table 2.

Table 2. Initial tier allocation for test cases with shared ASM disk groups

Databases	ASM disk groups	Thin devices	Storage Group	Thin Pool	RAID	Tier associated	Initial Tier Allocation
FINDB & HRDB	+Data	12 x 100 GB	DATA_SG	FC_Pool	RAID5	FC	100%
				EFD_Pool		EFD	0%
				SATA_Pool		SATA	0%
	+REDO	4 x 6 GB	REDO_SG	REDO_Pool	RAID1	FC	N/A[3]

One server was used for this test. Each of the Oracle databases was identical in size (about 600 GB) and designed for an industry-standard OLTP workload. However during this test one database had high activity whereas the other database remained idle to provide a simple example of the behavior of FAST VP.

Note that since an industry-standard benchmark tool was used, the I/O distribution across the database was completely even and random. This reduced sub-LUN skewing (since the whole database was highly active), and therefore the second idle database helped in simulating a more normal environment where some objects won't be highly accessed. It is very likely that real customer databases will demonstrate much better locality of data referenced (the recent data is more heavily accessed, or a mix of hot and cold database objects), providing FAST VP with better sub-LUN skewing to work with. With improved locality of reference (sub-LUN skewing) smaller EFD capacity can contain the hot database objects and therefore the policy can be set to a smaller EFD tier allocation percentage than shown in this example.

Test case execution

Objectives

Achieve a single Oracle ASM database workload storage tiering optimization by FAST VP.

Steps

1. Run a baseline workload prior to the FAST VP enabled run
2. Run the workload with FAST VP enabled, allowing storage allocation on all three tiers
3. Review the storage tiering efficiency and performance differences

Monitoring database and storage performance

During the baseline run the database devices were 100 percent allocated on the FC tier as shown in Table 3. Per the AWR report given in Table 4 user I/O random read

[3] The logs from both databases shared the ASM disk group +REDO, which was associated with the FC tier and not under FAST VP control.

activity (“db file sequential read”) is the main database wait event, with an average I/O response time of 6 ms. For FC drives this is a good response time that reflects a combination of 15k rpm drives (typically 6 ms response time at best per I/O, regardless of storage vendor) with efficient Symmetrix cache utilization.

Table 3. FINDB initial storage allocation

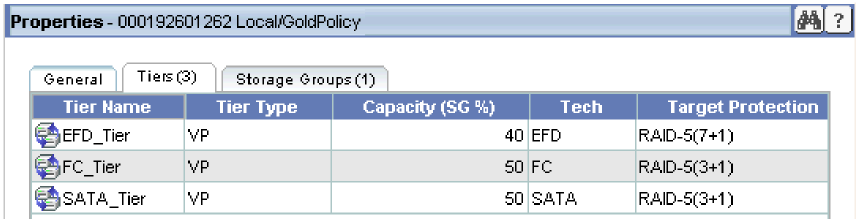
ASM disk group	Database Size	Initial Storage Tier Allocation		
+DATA (1.2 TB)	FINDB (600 GB) HRDB (600 GB)	EFD	0%	0
		FC	100%	1.2 TB
		SATA	0%	0

Table 4. Initial AWR report for FINDB

Event	Waits	Time(s)	Avg wait (ms)	% DB time	Wait Class
db file sequential read	3,730,770	12,490	6	88.44	User I/O
db file parallel read	85,450	1,249	14	6.74	User I/O
DB CPU		674		4.79	
log file sync	193,448	108	1	0.56	Commit
db file scattered read	3,241	20	11	0.22	User I/O

Defining the FAST VP policy

Although a 6 ms response time is very good for a FC tier with a heavy I/O workload, a FAST VP “Gold” policy was set to improve both the performance for this critical database as well to tier it across SATA, FC, and EFD thin pools. As shown in Figure 12, which is part of a Symmetrix Management Console (SMC) screen, the Gold policy allowed a maximum 40 percent allocation on the EFD tier[4] and 50 percent allocations on both of the FC and SATA tiers.



Tier Name	Tier Type	Capacity (SG %)	Tech	Target Protection
EFD_Tier	VP	40	EFD	RAID-5(7+1)
FC_Tier	VP	50	FC	RAID-5(3+1)
SATA_Tier	VP	50	SATA	RAID-5(3+1)

Figure 12. Gold FAST VP policy storage group association

[4] See the earlier note explaining why the specific database workload generator used required such high EFD allocation where it is expected that any true database workload will have much finer data access skewing and will get better benefits with less EFD.

Running the database workload after enabling the FAST VP policy

The database workload was restarted after enabling the FAST VP policy. FAST VP collected statistics, analyzed them, and performed the extent movements following the performance and compliance algorithms.

As can be seen in Figure 13, the tier allocation changed rapidly and where the FC tier was 100 percent used at the beginning of the run, by the end of the run the ASM disk group was using 35 percent of the EFD tier and rest of the disk group was spread across FC and SATA tiers. As the entire +DATA ASM disk group was associated with a FAST VP policy and FINDB and HRDB were sharing the same ASM disk group, the majority of active extents of FINDB moved to the EFD tier whereas inactive extents of HRDB moved to the SATA tier. The extents that were moderately active remained in the FC storage tier. At the end of the run the ASM disk group was spread across all three storage tiers based on the workload and FAST VP policy.

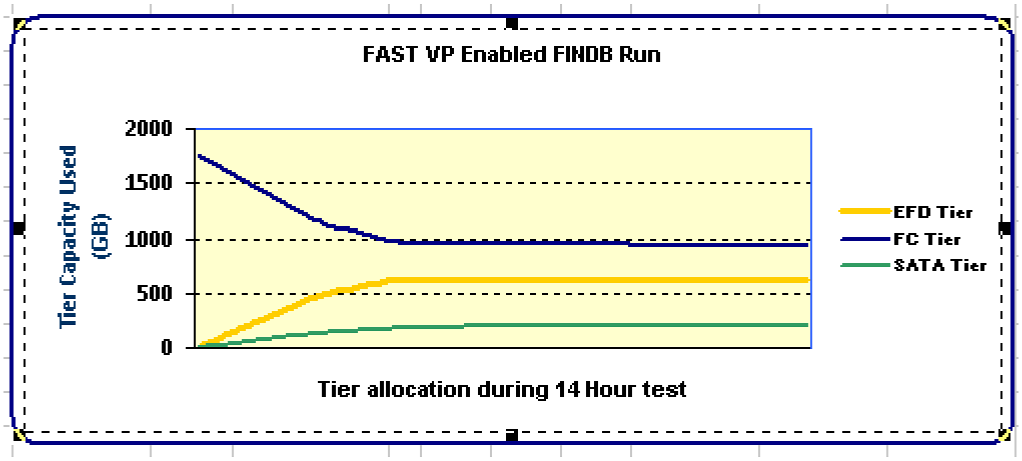


Figure 13. Storage tier allocation changes during the FAST VP enabled run

The storage tier allocations initially and after FAST VP was enabled are shown in Table 5. The Solutions Enabler command lines for enabling FAST VP operations and monitoring tier allocations are given in the Appendix on page 43.

Table 5. Oracle database tier allocations – Initial and FAST VP enabled

ASM disk group	Database Size	Storage Tiers Allocation				
		Tier Used	Initial		FAST VP Enabled	
+DATA (1.2 TB)	FINDB (600 GB) HRDB (600 GB)	EFD	0%	0	35%	626 GB
		FC	100%	1.2 TB	50%	941 GB
		SATA	0%	0	12%	204 GB

Analyzing the performance improvements with FAST VP

As can be seen in Table 6, the average I/O response time at the end of the run changed to 3 ms, which is a considerable improvement over the initial test that utilized the FC tier for the entire ASM disk group. This is the result of migration of active extents of the ASM disk group to EFD tiers and allocation of 35 percent capacity on that tier.

Table 6. FAST VP enabled database response time from the AWR report

Event	Waits	Time(s)	Avg wait (ms)	% DB time	Wait Class
db file sequential read	3,730,770	12,490	3	86.84	User I/O
db file parallel read	85,450	1,249	15	8.68	User I/O
DB CPU		674		4.69	
log file sync	193,448	108	1	0.75	Commit
db file scattered read	3,241	20	6	0.14	User I/O

The response time improvement and utilization of all available storage tiers — EFD, FC and SATA — to store ASM disk group extents also resulted in considerable improvement in FINDB transaction rates as shown in the next figure. The initial database transaction rate (transactions per minute) for FINDB with the entire ASM disk group on the FC tier was 2,079, and after FAST VP initiated movements a transaction rate of 3,760 was achieved that is *an improvement of **81 percent*** while utilizing all available storage tiers more effectively and efficiently.

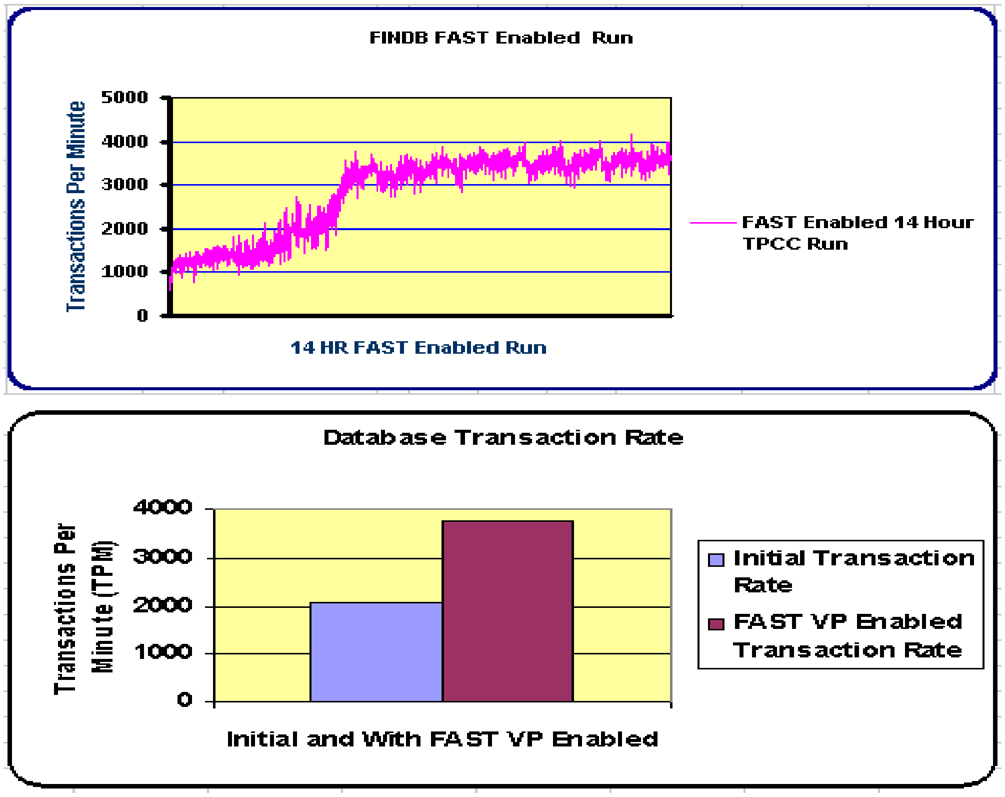


Figure 14. Database transaction rate changes with FAST VP

Test Case 2: Oracle databases sharing the ASM disk group and FAST policy

Oracle ASM makes it easy to provision and share devices across multiple databases. The databases, running different workloads, can share the ASM disk group for ease of manageability and provisioning. Multiple databases can share the Symmetrix thin pools for ease of provisioning, wide striping, and manageability at the storage level as well. This section describes the test case in which a FAST VP policy is applied to the storage group associated with the shared ASM disk group. At the end of the run we can see improved transaction rates and response times of both databases, and very efficient usage of the available tiers.

Test case execution

Objectives

Achieve storage tiering optimization for multiple databases sharing the ASM disk group using FAST VP.

Steps

1. Run performance baselines while both databases use the FC tier alone (prior to the FAST VP enabled run)
2. Run the workload again on both databases with FAST VP enabled, allowing storage allocation on all three tiers
3. Review the storage tiering efficiency and performance differences

Monitoring database and storage performance

During the baseline run the databases devices were 100 percent allocated on the FC tier as shown in Table 7. Both databases executed an OLTP-type workload (similar to the previous use case) where FINDB had more processes executing the workload in comparison to HRDB's workload, and therefore FINDB had a higher workload profile than HRDB.

Table 7. FINDB and HRDB initial storage allocation

ASM disk group	Database Size	Initial Storage Tier Allocation		
+DATA (1.2 TB)	FINDB (600 GB) HRDB (600 GB)	EFD	0%	0
		FC	100%	1.2 TB
		SATA	0%	0

Defining the FAST VP policy

As the ASM disk group and Symmetrix storage groups are identical to the ones used in Test Case 1 the same FAST policy is used for this use case.

Running the database workload after enabling the FAST VP policy

At the start of the test FAST VP was enabled and workloads on both databases started with FINDB running a higher workload compared to HRDB. After an initial analysis period (which was 2 hours by default) FAST performed the movement, and the tier allocation resulting from FAST VP-based movement can be seen in Figure 15.

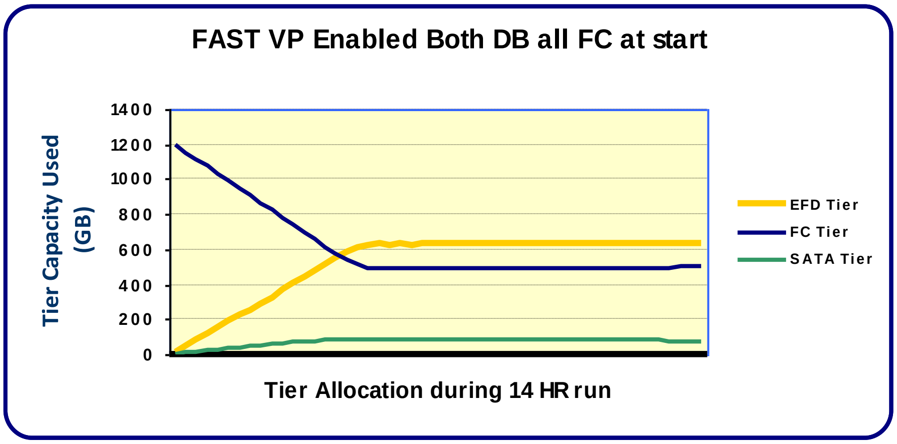


Figure 15. Storage tier allocation changes during the FAST VP-enabled run

Analyzing performance improvements with FAST VP

Active extents from both databases were distributed to the EFD and FC tiers with the majority of active extents on EFDs while inactive extents migrated to the SATA tier. Figure 16 shows the performance improvements for both databases, which are associated with the tier allocation changes as shown in Figure 15.

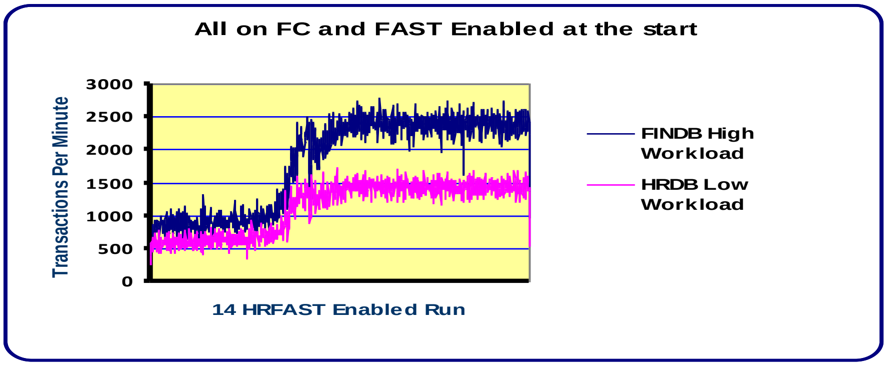


Figure 16. Storage tier allocation changes during the FAST VP-enabled run

The database transaction rate changes before and after FAST-based movements are shown in Table 8. Both databases exhibited higher performance with FINDB, which was more active and achieved higher gain as more extents from FINDB got migrated to EFDs.

Table 8. FAST VP-enabled transaction rate changes

Database	Transaction Rate		% Improvement
	Initial	FAST VP Enabled	
FINDB	1144	2497	118%
HRDB	652	1222	87%

Test Case 3: Oracle databases on separate ASM disk groups and FAST policies

Not all databases have the same I/O profile or SLA requirements and may also warrant different data protection policies. By deploying the databases with different profiles on separate ASM disk groups, administrators can achieve the desired I/O performance and ease of manageability. On the storage side these ASM disk groups will be on separate storage groups to allow for definition of FAST VP policies appropriate for the desired performance. This section describes a use case with two Oracle databases with different I/O profiles on separate ASM disk groups and independent FAST policies.

The hardware configuration of this test was the same as the previous two use cases (as shown in Table 1 on page 31). This test configuration had two Oracle databases — CRMDB (CRM) and SUPCHDB (Supply Chain) — on separate ASM disk groups, storage groups, and FAST VP policies, as shown in Table 9.

Table 9. Initial tier allocation for a test case with independent ASM disk groups

Databases	ASM Disk Group	Thin devices	Storage Group	Thin Pool	RAID	Tier associated	Initial Tier Allocation
CRMDB	+Data	6 x 100 GB	OraDevices_C1	FC_Pool	RAID 5	FC	100%
	+REDO	2 x 6 GB	OraRedo	EFD_Pool		EFD	
SUPCHDB	+Data	6 x 100 GB	OraDevices_S1	SATA_Pool		SATA	
	+REDO	2 x 6 GB	OraRedo	REDO_Pool	RAID 1	FC	N/A[5]

The Symmetrix VMAX array had a mix of storage tiers – EFD, FC, and SATA. One server was used for this test. Each of the Oracle databases was identical in size (about 600 GB) and designed for an industry-standard OLTP workload.

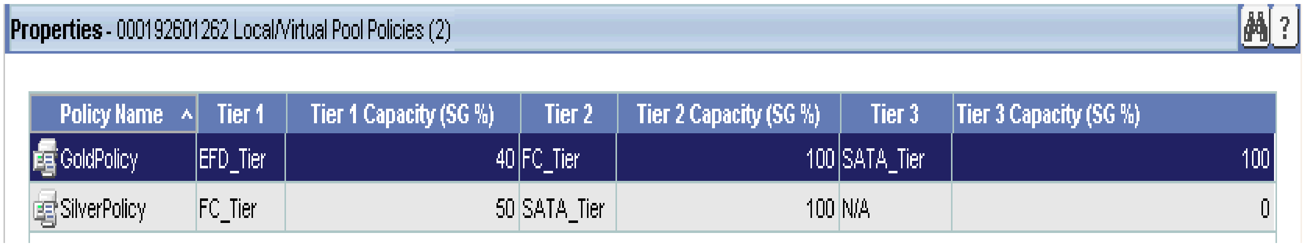
The Oracle databases CRMDB and SUPCHDB used independent ASM disk groups based on thin devices that were initially bound to FC_Pool (FC tier).

[5] As in the previous use cases, the redo logs were excluded from FAST VP policy to focus the tests on the data file's activity.

The CRMDB database in this configuration was part of a customer relationship management system that was critical to the business. To achieve higher performance the FAST VP policy “GoldPolicy” was defined to make use of all three available storage tiers, and storage group - OraDevices_C1 was associated with the policy.

The SUPCHDB database was important to the business and had proper performance characteristics. Business would benefit if the performance level can be maintained at lower cost. To meet this goal the FAST VP policy “SilverPolicy” was defined to make use of only FC and SATA tiers, and storage group - OraDevices_S1 was associated with the policy.

The FAST policies are shown in Figure 17.



Properties - 000192601262 Local/Virtual Pool Policies (2)

Policy Name	Tier 1	Tier 1 Capacity (SG %)	Tier 2	Tier 2 Capacity (SG %)	Tier 3	Tier 3 Capacity (SG %)
GoldPolicy	EFD_Tier	40	FC_Tier	100	SATA_Tier	100
SilverPolicy	FC_Tier	50	SATA_Tier	100	N/A	0

Figure 17. FAST Gold (CRMDB) and Silver (SUPCHDB) policies

Test case execution

Objectives

Achieve storage tiering optimization while maintaining isolation of resources that each database is allowed to use.

Steps

1. Run a baseline workload (prior to the FAST VP-enabled run)
2. Define two separate FAST policies – Gold policy and Silver policy – and associate them with the appropriate storage groups
3. Run the workloads again with FAST VP enabled, allowing storage allocation based on the distinct FAST VP policies
4. Review the storage tiering efficiency and performance differences

Monitoring database and storage performance

The following table shows the baseline performance of both databases based on the initial FC tier allocation. Both databases are getting a response time of 8 ms. Our goal is to improve it for CRMDB and maintain it for SUPCHDB at lower cost.

Table 10. Initial AWR reports for CRMDB and SUPCHDB

CRMDB					
Event	**Waits**	**Time(s)**	**Avg wait (ms)**	**% DB time**	**Wait Class**
db file sequential read	13,566,056	104,183	**8**	92.87	User I/O
db file parallel read	300,053	10,738	19	6.07	User I/O
DB CPU		4,338		2.45	
log file sync	1,635,001	1,157	1	0.65	Commit
db file scattered read	33,212	285	9	0.16	User I/O

SUPCHDB					
Event	**Waits**	**Time(s)**	**Avg wait (ms)**	**% DB time**	**Wait Class**
db file sequential read	8,924,638	69,515	**8**	92.93	User I/O
db file parallel read	194,525	5,774	19	4.9	User I/O
DB CPU		2,196		1.86	
log file sync	746,897	585	1	0.5	Commit
db file scattered read	17,860	208	12	0.18	User I/O

Defining the FAST VP policy

For CRMDB, our goal was to improve the performance. For FC-based configurations, a response time of 8 ms is reasonable, but can improve with better storage tiering. The FAST VP Gold policy was defined to improve both the performance for this critical database as well to tier it across SATA, HDD, and EFD thin pools. The Gold policy allowed a maximum 40 percent allocation on the EFD tier[6] and 100 percent allocations on both of the FC and SATA tiers. By setting FC and SATA allocations to 100 percent in this policy, FAST VP has the liberty to leave up to 100 percent of the data on any of these tiers or move up to 40 percent of it to EFD, based on the actual workload.

For SUPCHDB, our goal was to lower the cost while maintaining or improving the performance. The FAST VP Silver policy was defined to allocate the extents across FC and SATA drives to achieve this goal. The Silver policy allows a maximum of 50 percent allocation on the FC tier and up to 100 percent allocation on the SATA tier.

Running the database workload after enabling the FAST VP policy

The database workload was repeated after enabling the FAST VP policy. FAST VP collected statistics, analyzed them, and performed the extent movements following

[6] See the earlier note explaining why the specific database workload generator used required such high EFD allocation, where it is expected that actual customers' workloads will have much finer data access skewing and will get better benefits with less EFD.

the performance and compliance algorithms. The AWR reports for both databases were generated to review the I/O response times as shown in Table 11.

Table 11. FAST VP-enabled AWR reports for CRMDB and SUPCHDB

CRMDB

Event	Waits	Time(s)	Avg wait (ms)	% DB time	Wait Class
db file sequential read	32,332,795	160,945	**5**	91.04	User I/O
db file parallel read	720,608	10,738	15	6.07	User I/O
DB CPU		4,338		2.45	
log file sync	1,635,001	1,157	1	0.65	Commit
db file scattered read	33,212	285	9	0.16	User I/O

SUPCHDB

Event	Waits	Time(s)	Avg wait (ms)	% DB time	Wait Class
db file sequential read	15,035,122	109,502	**7**	92.87	User I/O
db file parallel read	328,884	5,774	18	4.9	User I/O
DB CPU		2,196		1.86	
log file sync	746,897	585	1	0.5	Commit
db file scattered read	17,860	208	12	0.18	User I/O

The database transaction rate changes are shown in Figure 18.

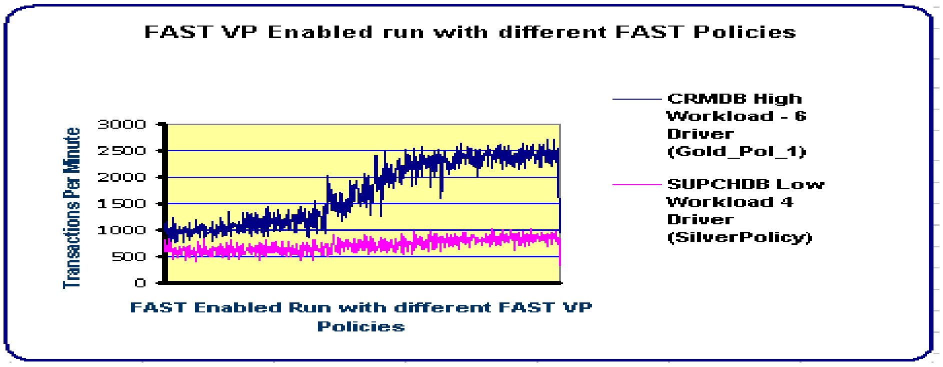


Figure 18. CRMDB and SUPCHDB transaction rate changes with FAST VP

Analyzing the performance improvements with FAST VP

As shown in Table 12, CRMDB used the FAST Gold policy and FAST VP migrated 40 percent of the CRMDB FC extents to the EFD tier and 10 percent to SATA. The rest of the extents remained on FC drives. This resulted in improvement of response time from 8 ms to 5 ms and a very decent improvement in transaction rate from 962 to

2,500, which represents 160 percent growth in transaction rate without any application change.

SUPCHDB used the FAST Silver policy and therefore FAST VP moved the less active extents to SATA drives. Still, the response time improved from 8 ms to 7 ms and hence we reached both cost savings while maintaining or improving performance.

Table 12. Storage tier allocation changes during the FAST VP-enabled run

ASM disk group	Database	DB Size	Initial Transaction Rate (TPM)	FAST VP Enabled Transaction Rate (TPM)	% Change	FAST Policy Used	FAST VP Enabled Storage Tiers Used		
							EFD	FC	SATA
+DATA	CRMDB	600 GB	962	2500	160%	GOLD Policy	40%	50%	10%
							240 GB	300 GB	60 GB
	SUPCHDB	600 GB	682	826	21%	Silver Policy		44%	56%
							0	266 GB	334 GB

Conclusion

Symmetrix Virtual Provisioning offers great value to Oracle environments with improved performance and ease of management due to wide striping and higher capacity utilization. Oracle ASM and Symmetrix Virtual Provisioning complement each other very well. With a broad range of data protection mechanisms and tighter integration between Symmetrix and Oracle now available even for thin devices, adoption of Virtual Provisioning for Oracle environments is very desirable.

Symmetrix FAST VP in Oracle environments improves storage utilization and optimizes the performance of databases by effectively making use of multiple storage tiers at a lower overall cost of ownership.

Appendix: Solutions Enabler command lines (CLI) for FAST VP operations and monitoring

This appendix describes the Solutions Enabler commands lines (CLI) that can be used to configure and monitor FAST VP operations. All such operations can also be executed using the GUI of SMC. Although there are command line counterparts for the majority of the SMC-based operations, the focus here is to show only some basic tasks that operators may want to use CLI for.

A. Enabling FAST

Operation: Enable or disable FAST operations.

Command:

```
symfast –sid ‹Symm ID› enable/disable
```

B. Gathering detailed information about a Symmetrix thin pool

Operation: Show the detailed information about a Symmetrix thin pool.

Command:

```
symcfg show –pool FC_Pool –sid <Symm ID> -detail –thin
```

Sample output:

```
Symmetrix ID                      : 000192601262
Pool Name                         : FC_Pool
Pool Type                         : Thin
Dev Emulation                     : FBA
Dev Configuration                 : RAID-5(3+1)
Pool State                        : Enabled
....
Enabled Devices(20): ← Number of Enabled Data Devices (TDAT)  in the Thin Pool
  {
  ------------------------------------------------------
  Sym      Total    Alloc      Free Full     Device
  Dev      Tracks   Tracks    Tracks  (%)    State
  ------------------------------------------------------
  00EA    1649988   701664    948324  42  Enabled
  00EB    1649988   692340    957648  41  Enabled
...
}
Pool Bound Thin Devices(20): ← Number of Bound Thin Devices (TDEV)  in the Thin Pool
  {
  -----------------------------------------------------------------------
                   Pool             Pool              Total
  Sym        Total  Subs        Allocated         Written
  Dev       Tracks  (%)     Tracks   (%)     Tracks   (%)  Status
  -----------------------------------------------------------------------
  0162      1650000    5    1010940   61    1291842   78     Bound
```

C. Checking distribution of thin device tracks across FAST VP tiers

Operation: Listing the distribution of thin device extents across FAST VP tiers that are part of a FAST VP policy associated with the storage group containing the thin devices.

Command:

symcfg –sid <Symm ID> list –tdev –range **0162:0171** –detail

Sample output:

```
Symmetrix ID: 000192601262
Enabled Capacity (Tracks) :  363777024
Bound   Capacity (Tracks) :  26400000
             SYMMETRIX THIN DEVICES
------------------------------------------------------------------------------
                                              Pool      Pool          Total
                         Flags    Total     Subs   Allocated     Written
Sym  Pool  Name      EM       Tracks   (%)   Tracks (%)    Tracks (%)    Status
---- ------------ ----- --------- ----- --------- --- --------- --- -----------
0162 FC_Pool       FX           1650000  5       1010940  61     1291842  78     Bound
       EFD_Pool      --          -     -                259212  16          -  - -
       SATA_Pool     --           -     -                  21732    1          -  - -

Shows that Symmetrix thin device 0162 has thin device extents spread across data devices on FC_Pool,
EFD_Pool and SATA_Pool


. . .


0171 FC_Pool       FX           1650000  5           3720     0     1505281  91     Bound
       EFD_Pool      --          -     -                    2040    0          -  - -
       SATA_Pool     --           -     -              1499184 91          -  - -



Legend:
 Flags:  (E)mulation : A = AS400, F = FBA, 8 = CKD3380, 9 = CKD3390
       (M)ultipool : X = multi-pool allocations, . = single pool allocation
```

D. Checking the storage tiers allocation

Operation: Listing the current allocation of defined storage tiers on a Symmetrix.

Command:

```
symtier list -vp
```

Sample output:

```
Symmetrix ID        : 000192601262

---------------------------------------------------------------------
                                               I  Logical Capacities (GB)
                            Target             n  --------------------------------
Tier Name               Tech Protection        c  Enabled    Free    Used
--------------------- ---- ------------ - -------- -------- ---------------------

EFD_Tier              EFD    RAID-5(7+1)  S    2566       2565        1
FC_Tier               FC      RAID-5(3+1) S    4028       2814     1214
SATA_Tier             SATA  RAID-5(3+1)  S    2566       1435     1131

Shows that Symmetrix has 3 tiers defined: EFD_Tier, FC_Tier and SATA_Tier and their associated
enabled, free and used capacities

Legend:
  Inc Type     :  S = Static, D = Dynamic
```